

WHAT IS CLAIMED IS:

1. A flash memory data structure, comprising:

2 fixed length cells, each having:

3 a control and identifier section for containing a unique
4 identifier and a cell count for logically associating multiple
5 of said fixed length cells, and

6 a data section for containing only a configuration value
7 pertaining to said unique identifier.

2. The data structure recited in Claim 1 wherein said unique

2 identifier is one byte long.

3. The data structure recited in Claim 1 wherein one of said

2 fixed length cells equals a minimum storage space for said
3 configuration value.

4. The data structure recited in Claim 1 wherein said fixed

2 length is determined based on optimizing storage space of said data
3 structure.

5. The data structure recited in Claim 1 wherein said fixed

2 length cells are 32 bytes long, said control and identifier section
3 is 4 bytes long and said data section is 28 bytes long.

6. The data structure recited in Claim 1 wherein said data
2 section is located at an end of said fixed length cells.

7. The data structure as recited in Claim 1 wherein a length
2 of said fixed length cells is configurable by a programming macro.

8. The data structure as recited in Claim 1 wherein a size
2 of said data structure is configurable by a programming macro based
3 on a manufacturing stage of development.

9. The data structure as recited in Claim 1 wherein said
2 unique identifier corresponds to a configuration parameter in a
3 lookup table.

10. The data structure as recited in Claim 1 wherein
2 multiples of said unique identifier correspond to greater than 254
3 configuration parameters.

11. The data structure as recited in Claim 1 wherein said
2 control and identifier section is configurable such that said
3 unique identifier and said cell count are located in subsequent
4 bytes at the beginning of said control and identifier section.

12. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 1.

13. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 2.

14. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 3.

15. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 4.

16. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 5.

17. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 6.

18. A flash memory controller for imposing on a flash memory
2 the data structure as recited in Claim 11.

19. A flash memory containing the data structure as recited
2 in Claim 1.

20. A flash memory containing the data structure as recited
2 in Claim 2.

21. A flash memory containing the data structure as recited
2 in Claim 3.

22. A flash memory containing the data structure as recited
2 in Claim 4.

23. A flash memory containing the data structure as recited
2 in Claim 5.

24. A flash memory containing the data structure as recited
2 in Claim 6.

25. A flash memory containing the data structure as recited
2 in Claim 11.

26. A method of writing to flash memory with fixed length
2 cells, comprising:

3 locating a first of said fixed length cells that is free;

4 writing a unique identifier in a control and identifier
5 section of said first free fixed length cell;

6 writing a configuration value pertaining to said unique
7 identifier in a data section of said first free fixed length cell;
8 and

9 updating a cell count in said control and identifier section
10 to represent a number of said fixed length cells logically
11 associated.

27. The method as recited in Claim 26 further including
2 locking interrupts and updating a checksum of said configuration
3 value in said control and identifier section.

28. The method as recited in Claim 26 further including
2 searching said flash memory for a pre-existing configuration value
3 having said unique identifier and marking said pre-existing
4 configuration value as deleted.

29. The method as recited in Claim 26 further including
2 updating a global variable during system initialization with an
3 address of a first of said fixed length cells that is free.

30. The method as recited in Claim 29 further including
2 testing said configuration value to determine completeness.

31. The method as recited in Claim 30 further including
2 updating said cell count and marking said configuration value
3 as deleted when determining said configuration value is not
4 complete; and

5 updating said cell count and a checksum of said configuration
6 value when determining said configuration value is complete.

32. The method as recited in Claim 31 further including
2 validating checksums of each of said fixed length cells.

33. A method of searching for data in flash memory with fixed
2 length cells, comprising:

3 locating a first of said fixed length cells that is free; and
4 locating said data by searching downward from said first free
5 fixed length cell to other fixed length cells having a lower
6 address thereof.

34. The method as recited in Claim 33 wherein said data is
2 configuration data.

35. The method as recited in Claim 33 wherein said data is
2 located in a data section at the beginning of said fixed length
3 cells.